

A New Perspective of Proximal Gradient Algorithms

Yi Zhou^{a,*}, Yingbin Liang^b, Lixin Shen^c

^aDepartment of Electrical Engineering, Syracuse University

^bDepartment of Electrical Engineering, Syracuse University

^cDepartment of Mathematics, Syracuse University

Abstract

We provide a new perspective to understand proximal gradient algorithms. We show that both proximal gradient algorithm (PGA) and Bregman proximal gradient algorithm (BPGA) can be viewed as generalized proximal point algorithm (GPPA), based on which more accurate convergence rates of PGA and BPGA are obtained directly. Furthermore, based on GPPA framework, we incorporate the back-tracking line search scheme into PGA and BPGA, and analyze the convergence rate with numerical verification.

Keywords: Bregman distance, convergence rate, generalized proximal point algorithms, proximal gradient algorithms

1. Introduction

Proximal algorithms have been extensively studied in non-smooth convex optimization problems. These algorithms have broad applications in practical problems including image processing, e.g., [1, 2], distributed statistical learning, e.g., [3], and low rank matrix minimization, e.g., [4] due to their simplicity. The key idea of proximal algorithms is to smooth the objective function via various smoothing techniques [5, 6]. For example, the popular proximal point algorithm (PPA) applies a smoothing technique based on Moreau envelope as shown below.

Consider an optimization problem as follows:

$$(\mathbf{P1}) \quad \min_{x \in \overline{C}} g(x), \quad (1)$$

where $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ is a proper, closed and lower semicontinuous convex function, and C is a non-empty open convex set in \mathbb{R}^n with its closure being \overline{C} . In [7], PPA was introduced, which generates a sequence $\{x_k\}$ via the following iteration step:

$$(\text{PPA}) \quad x_{k+1} = \operatorname{argmin}_{x \in \overline{C}} \left\{ g(x) + \frac{1}{2\lambda_{k+1}} \|x - x_k\|_2^2 \right\}, \quad (2)$$

where λ_k are positive numbers. It was shown in [8, 9] that for properly chosen parameters λ_k the sequence converges to a solution to **(P1)**. The convergence rate of PPA was shown in [10].

Various generalized proximal point algorithms (referred to as GPPA) have been proposed and analyzed, e.g., [11, 12, 13]. The idea is to replace the quadratic term $\frac{1}{2}\|x - x_k\|_2^2$ in PPA by $\Psi(x, x_k)$, where Ψ is a metric defined on

$\overline{C} \times C$ and does not necessarily take the quadratic form. As a consequence, the iteration step of GPPA reads as follows:

$$(\text{GPPA}) \quad x_{k+1} = \operatorname{argmin}_{x \in \overline{C}} \left\{ g(x) + \frac{1}{\lambda_{k+1}} \Psi(x, x_k) \right\}. \quad (3)$$

The most popular choices of Ψ are the Bregman distance in [14, 15] and the φ -divergence in [16]. By exploiting the structures of these distance-like metrics, the convergence rate of GPPA was established in [13, 14, 17]. A unified framework to analyze GPPA with various choices of distance metrics was proposed in [17].

Motivated by PPA, splitting algorithms have been designed and analyzed in [9, 18]. In particular, consider the following problem:

$$(\mathbf{P2}) \quad \min_{x \in \mathbb{R}^n} \{f(x) + g(x)\}, \quad (4)$$

where f is a differentiable convex function with a $1/\gamma$ Lipschitz continuous gradient, and g is a proper closed convex function. The proximal gradient algorithm (PGA) has been developed in [19] to solve **(P2)**, which has the iteration step given by

$$(\text{PGA}) : \quad x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ g(x) + \langle x, \nabla f(x_k) \rangle + \frac{1}{2\eta} \|x - x_k\|_2^2 \right\}, \quad (5)$$

where η is the step size satisfying $\eta \leq \gamma$. It has been shown in [1] that PGA has a convergence rate of $\mathcal{O}(1/k)^1$, and the rate can be further improved to be $\mathcal{O}(1/k^2)$ via techniques

*Corresponding author

Email address: yzhou35@syr.edu (Yi Zhou)

¹Here, $f(n) = \mathcal{O}(g(n))$ denotes that $|f(n)| \leq \xi|g(n)|$ for all $n > N$, where ξ is a constant and N is a positive integer.

of acceleration. Similarly, PGA can also be generalized by replacing the quadratic term, i.e., the third term in (5), with the Bregman distance. The corresponding algorithm is referred to as BPGA [20], and is given by

(BPGA) :

$$x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ g(x) + \langle x, \nabla f(x_k) \rangle + D_{\frac{1}{\eta}H}(x, x_k) \right\}, \quad (6)$$

where $D_{\frac{1}{\eta}H}$ is the Bregman distance (as defined in Definition 2 in Section 2) based on the convex function $\frac{1}{\eta}H$. It has been shown in [20] that the convergence rate of BPGA is $\mathcal{O}(1/k)$.

It is clear that PGA and BPGA are more general algorithms than the original PPA and GPPA (with Bregman distance), because they reduce to PPA and GPPA when f is a constant function. Thus, in existing literature, the convergence rates of PGA and BPGA are analyzed by their own as in [1, 20] without resorting to existing analysis of PPA and GPPA. In contrast to this conventional viewpoint of connections between PGA/BPGA and PPA/GPPA, the main contribution of this paper is to show that PGA and BPGA can, in fact, be viewed as GPPA with Bregman distance metrics, and thus both are special cases of GPPA. Consequently, the convergence rate of both algorithms can be obtained directly based on that of GPPA.

More specifically, we show that the sequences generated by PGA and BPGA are exactly the same as those generated by GPPA with Bregman distance metrics associated with properly chosen functions. This provides a new perspective that unifies PGA and BPGA into the framework of GPPA. To the best of our knowledge, such connection has not been established and reported in the existing literature. Consequently, the convergence rate of PGA and BPGA follows directly from that of GPPA, which is a much simpler way for convergence analysis than existing approaches. Interestingly, the convergence rate obtained in this way is more accurate than that obtained by directly analyzing PGA as in [1] and BPGA as in [20]. Moreover, based on the GPPA framework, we further incorporate the back-tracking line search scheme into PGA and BPGA, which is easier to implement in practice. We also analyze the convergence rate of such algorithms with line search and verify our analysis numerically.

The rest of the paper is organized as follows. In Section 2, we introduce necessary definitions and properties that are useful for our analysis. In Section 3, we present connections of PGA and BPGA to GPPA. In Section 4, we further develop PGA and BPGA with line search. Finally in Section 5, we conclude our paper with a few remarks on our results.

2. Preliminaries

In this section, we introduce definitions and properties that are useful for developing our results.

Definition 1 (Proximal Distance). *Let C be an open non-empty convex subset of \mathbb{R}^n . Let $\Psi : \mathbb{R}^n \times \mathbb{R}^n \rightarrow [0, +\infty]$ be a continuous function with $\operatorname{dom} \Psi(x, \cdot) = C$, $\operatorname{dom} \Psi(\cdot, y) = \overline{C}$ and $\operatorname{dom} \nabla_1 \Psi(\cdot, y) = C$ for every $x \in \overline{C}$ and $y \in C$. The function Ψ is said to be a proximal distance if the following conditions hold:*

- (a) $\Psi(\cdot, y)$ is differentiable and strictly convex for every $y \in C$;
- (b) $\Psi(\cdot, y)$ has bounded level set, i.e., for every $\alpha \in \mathbb{R}$, the set $\{x \in \overline{C} : \Psi(x, y) \leq \alpha\}$ for every $y \in C$ is bounded;
- (c) $\Psi(x, y) \geq 0$ for all $x \in \overline{C}$ and $y \in C$, and equality holds iff $x = y$.

Definition 2 (Bregman Distance). *Let C be an open non-empty convex subset of \mathbb{R}^n . Let $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ be proper with $\operatorname{dom} h = \overline{C}$. Assume h is continuous and strictly convex on \overline{C} , and continuously differentiable on C with $\operatorname{dom} \nabla h = C$. Then a distance metric D_h is a Bregman distance associated with function h if it is a proximal distance and has the following structure:*

$$D_h(x, y) = h(x) - h(y) - \langle x - y, \nabla h(y) \rangle, \quad (7)$$

for all $x \in \overline{C}$ and $y \in C$.

Note that (7) and the convexity of h imply the non-negativity of Bregman distance. Let D_h and $D_{h'}$ be two Bregman distances. Then, for all $a, b \in C$, and $c \in \overline{C}$, (7) implies that the following two properties hold:

- Three-point property:

$$D_h(c, a) + D_h(a, b) - D_h(c, b) = \langle \nabla h(b) - \nabla h(a), c - a \rangle. \quad (8)$$

- Linearity:

$$D_h(a, b) \pm D_{h'}(a, b) = D_{h \pm h'}(a, b). \quad (9)$$

One special GPPA scheme given in (3) is to set Ψ to be a Bregman distance D_h . In [17], the convergence rate of such GPPA with Bregman distance was established, which we present below for convenience.

Theorem 1 (Auslender and Teboulle [17], Theorem 2.1). *Let g be a proper, closed, lower semicontinuous convex function and let $D_h : \overline{C} \times C \rightarrow \mathbb{R}_+$ be a Bregman distance. Denote $\{x_k\}$ the sequence generated by GPPA with Bregman distance D_h . Assume $\operatorname{dom} g \cap \overline{C} \neq \emptyset$ and the solution set X^* of (P1) is non-empty. Set $\sigma_m = \sum_{k=1}^m \lambda_k$ with $\sigma_0 = 0$. Then for all $m > 0$ the following convergence rate holds:*

$$g(x_m) - g(x^*) \leq \frac{D_h(x^*, x_0)}{\sigma_m}, \quad \forall x^* \in X^*. \quad (10)$$

3. BPGA and PGA: A Special Case of GPPA

In this section, we first analyze BPGA for problem **(P2)**, and then specialize our result to its special case PGA. We first recall the iteration step of BPGA as follows for convenience:

(BPGA)

$$x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ g(x) + \langle x, \nabla f(x_k) \rangle + D_{\frac{1}{\eta}H}(x, x_k) \right\},$$

where η is the step size satisfying $\eta \leq \gamma$ with $1/\gamma$ being the Lipschitz constant of ∇f , and H is a differentiable and strongly convex function with $\operatorname{dom} H = \bar{C}$ and $\operatorname{dom} \nabla H = C$.

In the following, we reformulate BPGA as GPPA with a Bregman distance, based on which the convergence rate of BPGA can be established in a straightforward fashion.

Theorem 2. Consider problem **(P2)** and set $F = f + g$. Let H be differentiable and strongly convex with parameter $\sigma \geq \eta/\gamma$. Assume $\operatorname{dom} F \cap C \neq \emptyset$. Then for any initial seed $x_0 \in \operatorname{dom} F \cap C$, the sequence $\{x_k\}$ generated by BPGA is the same as the one generated by the following GPPA iteration:

$$x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \{F(x) + D_h(x, x_k)\}, \quad (11)$$

where D_h is Bregman distance based on function $h = \frac{1}{\eta}H - f$.

Proof. We first show that $h := \frac{1}{\eta}H - f$ is convex. To this end, it suffices to show that ∇h is a monotone operator. Indeed, for any $x, y \in \operatorname{dom} f \cap C$, the strong convexity of H and the Lipschitz continuity of ∇f yield

$$\begin{aligned} \langle \nabla H(x) - \nabla H(y), x - y \rangle &\geq \sigma \|x - y\|_2^2, \\ \langle \nabla f(x) - \nabla f(y), x - y \rangle &\leq \frac{1}{\gamma} \|x - y\|_2^2. \end{aligned}$$

Since $\sigma \geq \eta/\gamma$, the above two inequalities imply that

$$\langle \nabla h(x) - \nabla h(y), x - y \rangle \geq 0.$$

Hence, h is convex.

Due to the linearity (9) and the definition of Bregman distance (7), we obtain

$$\begin{aligned} F(x) + D_{\frac{1}{\eta}H-f}(x, x_k) &= f(x) + g(x) + D_{\frac{1}{\eta}H}(x, x_k) - D_f(x, x_k) \\ &= g(x) + \langle x, \nabla f(x_k) \rangle + D_{\frac{1}{\eta}H}(x, x_k) \\ &\quad + f(x_k) - \langle x_k, \nabla f(x_k) \rangle. \end{aligned}$$

Thus, x_{k+1} generated from BPGA is identical to the one generated by (11). This completes the proof. \square

Remark 1. The constraint $\eta \leq \gamma$ in BPGA guarantees h to be a convex function. By choosing smaller η , the function H can be less strongly convex. Since η can be chosen arbitrarily close to zero, the range of σ in Theorem 2 can be as large as $\sigma > 0$.

Since BPGA can be formulated as a special case of GPPA, the bound for GPPA in equation (10) applies and we directly obtain the following convergence rate for BPGA.

Corollary 1. Let $F = f + g$ and $h = \frac{1}{\eta}H - f$. Assume $\operatorname{dom} F \cap C \neq \emptyset$. Then for all $x^* \in X^*$, the sequence $\{x_k\}$ generated by BPGA satisfies the following convergence rate:

$$F(x_k) - F(x^*) \leq \frac{D_h(x^*, x_0)}{k}. \quad (12)$$

Proof. The proof follows by applying Theorem 2, and identifying g , Ψ , h , λ_k , and σ_k in equation (10) as F , D_h , $\frac{1}{\eta}H - f$, 1, and k , respectively. \square

We note that if $H = \frac{1}{2}\|\cdot\|_2^2$, then BPGA reduces to PGA. Hence the following corollary holds:

Corollary 2. Denote $F = f + g$ and $h = \frac{1}{2\eta}\|\cdot\|_2^2 - f$. Then for any $x^* \in X^*$, the sequence $\{x_k\}$ generated by PGA satisfies the following convergence rate:

$$F(x_k) - F(x^*) \leq \frac{D_{\frac{1}{2\eta}\|\cdot\|_2^2-f}(x^*, x_0)}{k},$$

for all $k > 0$.

We note that the convergence rate for PGA in Corollary 2 (referred to as GPPA-PGA rate) is more accurate than the one in [1] (referred to as PGA rate). We compare these two rates as follows:

$$\begin{aligned} \text{(GPPA-PGA rate)} & \frac{\frac{1}{2\eta}\|x^* - x_0\|_2^2 - D_f(x^*, x_0)}{k}, \\ \text{(PGA rate)} & \frac{1}{2\eta k}\|x^* - x_0\|_2^2. \end{aligned}$$

Although both rates are in the order of $\mathcal{O}(1/k)$, it is clear that the GPPA-PGA rate is more accurate. We next provide an example to illustrate this fact.

Consider **(P2)** with $f = \frac{1}{2\gamma}\|x - b\|_2^2$, and $g = \|x\|_1$, where b is a constant vector in \mathbb{R}^n . It is clear that the Lipschitz constant of ∇f is $\frac{1}{\gamma}$. Hence, by choosing $\eta = \gamma$, the iteration step of PGA is given by (after further simplification):

$$x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ \|x\|_1 + \frac{1}{2\gamma}\|x - b\|_2^2 \right\}. \quad (13)$$

That means that the algorithm converges in one step. Applying the GPPA-PGA rate with $k = 1$ and denoting $F = f + g$, we have that $F(x_1) - F(x^*) \leq$

$\frac{1}{2\gamma}\|x^* - x_0\|_2^2 - D_f(x^*, x_0) = 0$. We conclude that $F(x_1) = F(x^*)$, and hence the algorithm does converge in one step. The PGA rate is clearly not tight here. Therefore, this example demonstrates that we establish a more accurate convergence rate of PGA by the new perspective of PGA as GPPA with a special Bregman distance metric. Similarly, the convergence rate of BPGA given in Corollary 1 is in general tighter than the one in [20].

4. BPGA with Line Search

It can be observed that implementation of BPGA requires the knowledge of the Lipschitz constant γ in advance. If γ is unknown a priori, we incorporate the idea of line search into BPGA, and adaptively set Lipschitz constant for each iteration step. More specifically, iteration step k of BPGA with line search is given as follows:

- Set $x_{k+1} = \operatorname{argmin}_{x \in \mathbb{R}^n} \{F(x) + D_{h_k}(x, x_k)\}$ with $h_k = \frac{1}{\eta_k}H - f$;
- If $D_{h_k}(x_{k+1}, x_k) \geq 0$, set $h_{k+1} = h_k$ and go to iteration step $k+1$; Otherwise, set $\eta_k \leftarrow \alpha\eta_k$, where $\alpha < 1$, and repeat iteration step k .

We note that the line search criterion $D_{h_k}(x_{k+1}, x_k) \geq 0$ can be intuitively understood as at iteration k , we search for D_{h_k} that behaves as a distance between x_{k+1} and x_k .

The convergence rate of the BPGA with line search is characterized in the following theorem.

Theorem 3. *Let $F = f + g$ and $h = \frac{1}{\eta}H - f$. Assume $\operatorname{dom} F \cap C \neq \emptyset$. Then for all $x^* \in X^*$, the sequence $\{x_k\}$ generated by BPGA with line search as described above satisfies the following convergence rate:*

$$F(x_{m+1}) - F(x^*) \leq \frac{D_H(x_*, x_0)}{\alpha\gamma(m+1)}. \quad (14)$$

Proof. The idea of the proof follows that in [1]. We provide the proof here for the completeness of the paper. We first note that the backtracking line search scheme guarantees $\alpha\gamma \leq \eta_k$.

At iteration k , the argmin operation implies

$$F(x_{k+1}) + D_{h_k}(x_{k+1}, x_k) \leq F(x_k),$$

which combined with the line search criterion guarantees $F(x_{k+1}) < F(x_k)$, i.e., a descent method. By convexity of f and g , and set $g_{k+1} \in \partial g(x_{k+1})$, $F_{k+1} \in \partial F(x_{k+1})$. Then for any $x^* \in X^*$, we have

$$\begin{aligned} F(x^*) &\geq f(x_k) + \langle x^* - x_k, \nabla f(x_k) \rangle \\ &\quad + g(x_{k+1}) + \langle x^* - x_{k+1}, g_{k+1} \rangle \\ &= F(x_{k+1}) + \langle x^* - x_{k+1}, F_{k+1} \rangle \\ &\quad + D_f(x^*, x_{k+1}) - D_f(x^*, x_k). \end{aligned} \quad (15)$$

Furthermore, by the optimality condition of iteration $k+1$, we can choose F_{k+1} as

$$F_{k+1} = \nabla h_k(x_k) - \nabla h_k(x_{k+1}) \in \partial F(x^{k+1}).$$

Substituting the above expression into (15) and applying three-point property of Bregman distance, then with certain simplification we obtain

$$\begin{aligned} F(x^*) - F(x_{k+1}) &\geq \frac{1}{\eta_k} [D_H(x_*, x_{k+1}) - D_H(x_*, x_k)] \\ &\geq \frac{1}{\alpha\gamma} [D_H(x_*, x_{k+1}) - D_H(x_*, x_k)]. \end{aligned}$$

Taking the sum of the above inequality over $k = 0, \dots, m$ and applying the fact that $F(x_{k+1}) < F(x_k)$, then with certain simplification we obtain the desired result (14). \square

Remark 2. *For the PGA with $F = f + g$ and $h_k = \frac{1}{2\eta_k}\|\cdot\|_2^2 - f$, the line search condition $D_{h_k}(x_{k+1}, x_k) \geq 0$ reduces to the back-tracking line search in [1].*

We next verify the convergence rate of BPGA with line search via a numerical experiment. Consider the following problem

$$\operatorname{argmin}_{x \in \mathbb{R}^n} f(x) = \frac{1}{2}\|Ax - b\|_2^2, \quad \text{s.t. } x \in \Delta,$$

where $\Delta := \{x \in \mathbb{R}^{1000} : \sum_{j=1}^{1000} x_j = 1, x \geq 0\}$ is the simplex constraint set. We generate one realization of $A \in \mathbb{R}^{500 \times 1000}$ and $b \in \mathbb{R}^{500}$ from the normal distribution, then normalize b and columns of A to unit length. We solve the problem by two BPGA formulations. In the first formulation, we set $h_k(x) = \frac{1}{2\eta_k}\|x\|_2^2 - f(x)$, and set g to be the indicator function of the simplex set. In this case, BPGA reduces to PGA. In the second formulation, we set $h_k(x) = \frac{1}{\eta_k} \sum_{i=1}^n x_i \ln x_i - f(x)$, and set g to be the indicator function of the simplex set. In this case, BPGA corresponds to the mirror descent method [21]. We apply the line search criterion $D_{h_k}(x_{k+1}, x_k) \geq 0$ to both formulations and set $\eta_0 = 100$ with the decay rate $\alpha = 0.5$. We compare both methods with line search to the corresponding methods with the constant stepsize $\eta = 1/\lambda_{\max}(A^\top A)$. We initialize the algorithms with $x_0 = 10^{-3}\mathbf{1}_{1000}$, where $\mathbf{1}_{1000}$ denotes the 1000-dimensional vector with all entries being one.

Figure 1 demonstrates the convergence behavior of the four algorithms as the number of iterations changes. Four curves are plotted in this figure. The curves marked by “ \diamond ” and “ \bullet ” are generated by PGA and the mirror descent method with constant stepsize, respectively. Accordingly, the curves marked by “+” and “ \times ”, which mostly coincide with each other, are generated by PGA and the mirror descent method with line search, respectively. It can be observed that the line search scheme achieves faster convergence compared to the corresponding algorithm with constant stepsize. This is because a larger stepsize is used at each iteration of the line search algorithms. In particular, line search significantly improves the convergence rate of the mirror decent algorithm.

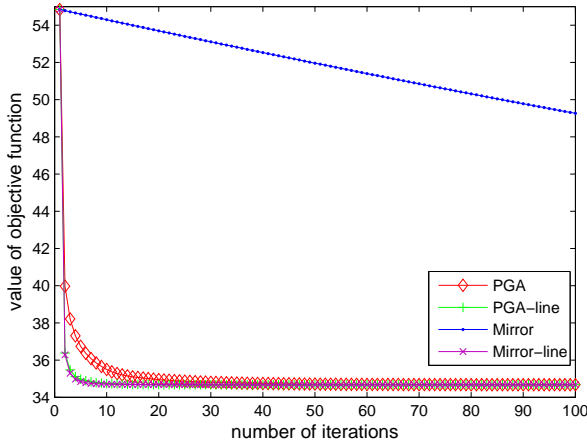


Figure 1: Comparison of PGA and mirror descent with line search to those with constant stepsize.

5. Conclusion

In this paper, we provided a new perspective of PGA and BPGA, and showed that both algorithms can be viewed as GPPA with special choices of Bregman distance. As a consequence, a more accurate convergence rate was established in a straightforward way for both PGA and BPGA. This new perspective sheds light on the essence of PGA: by properly choosing the Bregman distance in the GPPA scheme, the smooth part f is linearized, and hence the iteration leads to evaluate the proximity operator of g .

Acknowledgements

The work of Y. Zhou and Y. Liang was supported by the National Science Foundation under Grant CNS-11-16932. The work of L. Shen was supported in part by the National Science Foundation under grant DMS-11-15523 and by the National Research Council.

References

- [1] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM Journal of Image Science*. 2 (1) (2009) 183–202.
- [2] C. A. Micchelli, L. Shen, Y. Xu, Proximity algorithms for image models: Denoising, *Inverse Problems* 27 (2011) 045009(30pp).
- [3] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning* 3 (1).
- [4] B. Recht, M. Fazel, P. A. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization, *SIAM Review* 52 (3) (2010) 471–501.
- [5] Y. Nesterov, Smooth minimization of non-smooth functions, *Mathematical Programming* 103 (1) (2005) 127–152.
- [6] A. Beck, M. Teboulle, Smoothing and first order methods: A unified framework, *SIAM Journal on Optimization* 22 (2) (2012) 557–580.

- [7] B. Martinet, Brève communication. régularisation d'inéquations variationnelles par approximations successives, *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* 4 (R3) (1970) 154–158.
- [8] R. Rockafellar, Monotone operators and the proximal point algorithm, *SIAM Journal on Control and Optimization* 14 (5) (1976) 877–898.
- [9] J. Eckstein, D. P. Bertsekas, On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators, *Mathematical Programming* 55 (1992) 293–318.
- [10] O. Güler, On the convergence of the proximal point algorithm for convex minimization, *SIAM J. Control Optimization* 29 (2) (1991) 403–419.
- [11] Y. Censor, S. A. Zenios, Proximal minimization algorithm with d-functions, *Journal of Optimization Theory and Applications* 73 (3) (1992) 451–464.
- [12] M. Teboulle, Entropic proximal mappings with applications to nonlinear programming, *Mathematics of Operations Research* 17 (3) (1992) 670–690.
- [13] M. Teboulle, Convergence of proximal-like algorithms, *SIAM Journal on Optimization* 7 (4) (1997) 1069–1083.
- [14] G. Chen, M. Teboulle, Convergence analysis of a proximal-like minimization algorithm using Bregman functions, *SIAM Journal on Optimization* 3 (3) (1993) 538–543.
- [15] J. Eckstein, Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming, *Mathematics of Operations Research* 18 (1) (1993) 202–226.
- [16] A. N. Iusem, B. F. Svaiter, M. Teboulle, Entropy-like proximal methods in convex programming, *Mathematics of Operations Research* 19 (4) (1994) 790–814.
- [17] A. Auslender, M. Teboulle, Interior gradient and proximal methods for convex and conic optimization, *SIAM Journal on Optimization* 16 (3) (2006) 697–725.
- [18] P. Lions, B. Mercier, Splitting algorithms for the sum of two nonlinear operators, *SIAM Journal on Numerical Analysis* 16 (6) (1979) 964–979.
- [19] A. A. Goldstein, Convex programming in Hilbert space, *Bulletin of the American Mathematical Society* 70 (5) (1964) 709–710.
- [20] P. Tseng, Approximation accuracy, gradient methods, and error bound for structured convex optimization, *Mathematical Programming* 125 (2) (2010) 263–295.
- [21] A. Beck, M. Teboulle, Mirror descent and nonlinear projected subgradient methods for convex optimization, *Operations Research Letters* 31 (3) (2003) 167 – 175.